

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS

Sistemas de Informação

Letícia Thaís da Silva

**UMA PROPOSTA DE AUTOMAÇÃO RESIDENCIAL UTILIZANDO A
PLATAFORMA DE PROTOTIPAGEM ELETRÔNICA ARDUINO**

Betim

2013

Letícia Thaís da Silva

**UMA PROPOSTA DE AUTOMAÇÃO RESIDENCIAL UTILIZANDO A
PLATAFORMA DE PROTOTIPAGEM ELETRÔNICA ARDUINO**

Monografia apresentada ao Curso de
Sistemas de Informação da Pontifícia
Universidade Católica de Minas
Gerais, como requisito parcial para
obtenção do título de Bacharel em
Sistemas de Informação.

Orientador: Bernardo Jeunon

Betim

2013

Letícia Thaís da Silva

**UMA PROPOSTA DE AUTOMAÇÃO RESIDENCIAL UTILIZANDO A
PLATAFORMA DE PROTOTIPAGEM ELETRÔNICA ARDUINO**

Monografia apresentada ao Curso de
Sistemas de Informação da Pontifícia
Universidade Católica de Minas
Gerais, como requisito parcial para
obtenção do título de Bacharel em
Sistemas de Informação.

Bernardo Jeunon de Alencar, D. Sc. – ICEI/DCC PUC Minas

Leônidas Conceição Barros – PUC Minas

Betim, 04 de dezembro de 2013.

Aos meus pais,
pela paciência e por estarem sempre presentes me incentivando.

AGRADECIMENTOS

Agradeço primeiramente a Deus, por ter me concedido força, inteligência e sabedoria.

Agradeço aos meus pais por estarem sempre presentes me incentivando.

Agradeço ao professor Leonardo Grandinetti pelo incentivo no início do projeto, a professora Ester Jeunon por estar sempre disponível a ajudar, ao professor e orientador Bernardo Jeunon por todas ajudas, dicas e auxílios, e em especial ao professor Caio Veloso, que me ajudou muito durante esse trabalho, mas infelizmente não está mais entre nós.

Agradeço aos meus colegas acadêmicos pelos anos de companheirismo e amizade.

Agradeço à Thâmara Luiza por me ajudar na elaboração do logotipo do protótipo.

Agradeço ao meu irmão Guilherme Silva, por me ajudar na compreensão dos componentes elétricos necessários no projeto, e pela ajuda na confecção da maquete.

RESUMO

O presente trabalho tem como objetivo apresentar uma proposta de automação residencial, utilizando a plataforma de prototipagem eletrônica *open-source* Arduino. Ele apresenta um projeto para controle de iluminação, abertura e fechamento de portas, controle de alarmes e de câmera de segurança de uma residência fictícia.

O protótipo desenvolvido utiliza uma placa “Arduino Mega”, um módulo *Ethernet*, que será o responsável por prover conexão com a Internet ao Arduino, um servidor web Apache, que possui a função de disponibilizar as páginas do sistema web, o banco de dados MySQL, um dispositivo com acesso à internet e uma maquete residencial para a representação da casa modelo.

Por intermédio da aplicação web, desenvolvida através da linguagem PHP, o usuário irá escolher o comando que deseja executar que, enviados ao Microprocessador Arduino, executará a ação determinada.

Palavras chave: Arduino. Automação Residencial.

ABSTRACT

This paper aims to present a proposal for home automation using the platform open-source Arduino electronics prototyping. It presents a design for lighting control, opening and closing doors, alarm control and security camera of a fictitious residence.

The prototype uses a board "Arduino Mega", an Ethernet module, which is responsible for providing Internet connection to Arduino, an Apache web server, which has the function of providing the web pages of the system, the MySQL database a device with internet access and a residential model for the representation of the model home.

Through the web application, developed by the PHP language, the user will choose the command you want to run that sent the Arduino microprocessor, perform the given action.

Keywords: Arduino. Home Automation.

LISTA DE FIGURAS

Figura 1 – História da Domótica.....	17
Figura 2 - Ciclo de Adoção de Novos Produtos Pelos Consumidores	18
Figura 3 – Walpad IHouse.....	21
Figura 4 – Controle de sala GDS Automação	22
Figura 5 – Interação entre componentes do sistema.....	23
Figura 6 – Ambiente de desenvolvimento do Arduino.....	26
Figura 7 – Comunicação SPI	27
Figura 8 – Pinos utilizados para comunicação SPI	28
Figura 9 – Trecho de código chamada da biblioteca Ethernet.....	28
Figura 10 – Trecho de código utilizando biblioteca Servo	29
Figura 11 – Pino onde o servomotor é conectado	29
Figura 12 – Utilização do comando Write.....	30
Figura 13 – Arduino Mega.....	30
Figura 14 – Pinos Arduino	31
Figura 15 – Pinos do Arduino Mega 2560.....	33
Figura 16 – Módulo Ethernet	35
Figura 17 – Configurações do Arduino	35
Figura 18 – Princípio de funcionamento de um servo motor	36
Figura 19 – Funcionamento PHP, servidor Web e banco de dados.....	38
Figura 20 – Página inicial do PHPMyadmin	42
Figura 21 – Ligação de lâmpada sem automação	45
Figura 22 – Ligação de uma lâmpada ao quadro elétrico com quadro de automação	45
Figura 23 – Planta da maquete.....	46
Figura 24 – Sistema sendo acessado através de um smartphone	47
Figura 25 – Página Principal Domus Automação Residencial	47
Figura 26 – Página de gerenciamento de usuários	49
Figura 27 – Função socket_connect.....	50
Figura 28 – Mensagem de erro ao receber status da casa	50
Figura 29 – Tela de Controle do sistema.....	51
Figura 30 – Código mensagens enviadas pela página PHP	52

Figura 31 – Vetor declarado no código fonte do Arduino.....	52
Figura 32 – Comando enviado pela página PHP para abrir portão	52
Figura 33 – Comando enviado pela página PHP para fechar portão	53
Figura 34 – Programação servo motor	53
Figura 35 – Página de visualização de imagens	54

LISTA DE SIGLAS E ACRÔNIMOS

AURESIDE - Associação Brasileira de Automação Residencial

CSV - *Comma Separated Values*

HTML - *Hypertext Preprocessor*

ICSP - *In-Circuit Serial Programming*

IDE - *Integrated Development Environment*

IP - *Internet Protocol*

ISO - *Image System Optic*

LED - *Light Emitting Diode*

MAC - *Media Access Control*

MySQL – *My Structured Query Language*

PDF - *Portable Document Format*

PHP - *Hypertext Preprocessor*

PWM - *Pulse-Width Modulation*

QA – Quadro de automação

RJ45 - *Registered Jack 45*

SPI - *Serial Peripheral Interface*

TCP - *Transmission Control Protocol*

TI – Tecnologia da Informação

UDP - *User Datagram Protocol*

XML - *Extensible Markup Language*

SUMÁRIO

1 INTRODUÇÃO	13
1.1 Problematização.....	13
1.2 Objetivo geral	14
1.3 Objetivos específicos	14
1.4 Justificativa	14
1.5 Resultado esperado.....	14
1.6 Estrutura do trabalho.....	15
2 REVISÃO DE LITERATURA	16
2.1 História da Domótica	16
2.2 Desafios da Domótica.....	17
2.3 Características de sistemas de domótica.....	18
2.4 Benefícios proporcionados por sistemas de domótica.....	19
2.5 Mercado da domótica	20
2.6 Sistemas Comercializados	20
2.6.1 <i>IHouse</i>	20
2.6.2 <i>GDS Automação</i>	21
3 COMPONENTES FÍSICOS DO PROTÓTIPO.....	23
3.1 Plataforma Arduino	24
3.1.1 <i>O que é o Arduino?</i>	24
3.1.2 <i>História do Arduino</i>	24
3.1.3 <i>Ambiente de Desenvolvimento</i>	25
3.1.4 <i>Bibliotecas</i>	26
3.1.4.1 <u>Biblioteca SPI</u>	26
3.1.4.2 <u>Biblioteca Ethernet</u>	28
3.1.4.3 <u>Biblioteca Servo</u>	29
3.1.5 <i>Arduino Mega</i>	30
3.1.5.1 <u>Características</u>	30
3.1.5.2 <u>Função dos pinos</u>	31

3.2 Módulo Ethernet W5100	34
3.3 Servo motor	35
3.4 Câmera IP	36
4 TECNOLOGIAS PARA DESENVOLVIMENTO DO SISTEMA WEB	38
4.1 PHP.....	39
4.1.1 <i>História do PHP</i>	39
4.1.2 <i>Vantagens da PHP</i>	40
4.2 Apache	40
4.3 MySQL.....	41
5 DESENVOLVIMENTO DO PROTÓTIPO	44
5.1 O que muda ao Automatizar uma residência?	44
5.2 Desenvolvimento da maquete.....	46
5.3 Desenvolvimento da página Web	46
5.3.1 <i>Página inicial</i>	47
5.3.1.1 <u>Segurança da página</u>	48
5.3.2 <i>Página de gerenciamento de usuários</i>	49
5.3.3 <i>Página de controle do sistema</i>	49
5.3.3.1 <u>Botão de controle das lâmpadas</u>	51
5.3.3.2 <u>Botões de controle do portão</u>	52
5.3.3.3 <u>Botões de visualização das imagens da câmera</u>	53
5.3.4 <i>Página de visualização de imagens da câmera</i>	54
6 CONCLUSÃO E TRABALHOS FUTUROS	55
6.1 Trabalhos futuros.....	56
REFERÊNCIAS.....	57

1 INTRODUÇÃO

Nos últimos anos verifica-se que a automação está cada vez mais presente em ambientes domiciliares, proporcionando maior comodidade, segurança e confiabilidade. Isso consequentemente levou ao surgimento de uma nova área da automação, a domótica. A domótica é uma tecnologia recente, em que um sistema integrado é capaz de controlar a temperatura, luminosidade, nível de som, sistemas de segurança, entre outros, por meio de uma central de controle (BOLZANI, 2004).

A área de conhecimento se mostra propícia para atuação de um profissional de TI e revela alto potencial empreendedorístico.

Normalmente uma nova tecnologia possui custo mais elevado. Neste sentido diversas empresas buscam minimizar esse impacto sobre seus produtos através do uso de tecnologias *open source*, cujas licenças são distribuídas de forma gratuita. Este é o caso da plataforma de prototipagem eletrônica Arduino.

Assim, o objetivo deste trabalho é desenvolver um sistema de automação residencial baseado nesta plataforma.

1.1 Problemática

Por se tratar de uma tecnologia nova e pouco difundida, os sistemas de automação residencial de grandes empresas do ramo, tendem a possuir um alto custo de implementação. De acordo com a AURESIDE – Associação Brasileira de Automação Residencial – o custo de um sistema como o proposto representa 5% do valor de um imóvel (LANG, 2012), o que limita este mercado a pessoas com maior poder aquisitivo, tornando-se inatingível para boa parcela da população.

Verificou-se, assim, a oportunidade de realizar uma pesquisa e desenvolver um protótipo para conhecer, analisar e avaliar a possibilidade de se ter um sistema de automação residencial utilizando o Arduino, que, por se tratar de uma plataforma *open source*, poderia reduzir o custo de tal sistema.

1.2 Objetivo geral

O objetivo deste trabalho é desenvolver um sistema de automação residencial com algumas funções básicas baseado na plataforma de prototipagem eletrônica Arduino.

1.3 Objetivos específicos

- Identificar as potencialidades do Arduino para a automação residencial;
- Identificar os principais modelos e possibilidades de automação residencial;
- Desenvolver uma aplicação em PHP para acesso ao sistema;
- Desenvolver a programação da plataforma de prototipagem eletrônica Arduino;
- Construir uma maquete que represente uma residência automatizada.

1.4 Justificativa

O desenvolvimento desse trabalho justifica-se pela identificação de um novo nicho do mercado de automação residencial. Como já citado, os sistemas já existentes, possuem alto valor econômico, portanto deseja-se verificar se possível prover sistemas de automação através da plataforma Arduino, o que poderia reduzir este custo.

Outro fator relevante é o conhecimento que se pode agregar sobre a plataforma de prototipagem eletrônica Arduino, frequentemente utilizada em inúmeros projetos, das formas mais variadas.

1.5 Resultado esperado

Demonstrar como é possível desenvolver um sistema de automação, baseado na plataforma de código livre Arduino, capaz de prover segurança, praticidade e conforto para moradores de uma residência automatizada.

1.6 Estrutura do trabalho

Além deste capítulo introdutório essa monografia está estruturada em mais 5 capítulos, que estão dispostos da seguinte forma:

Capítulo 2 – Revisão de Literatura – Será apresentado a definição de alguns conceitos relacionado ao tema domótica.

Capítulo 3 – Componentes Físicos do Protótipo – Será apresentado os componentes físicos utilizados para a construção do protótipo.

Capítulo 4 – Tecnologias para Desenvolvimento do Sistema Web – Apresentará as ferramentas de desenvolvimento utilizadas para construção das páginas Web.

Capítulo 5 – Desenvolvimento do Protótipo – apresentar o que foi desenvolvido.

Capítulo 6 – Conclusão e trabalhos futuros – Trata-se da conclusão do projeto, e uma análise de quais melhorias podem ser aplicadas para a construção de um futuro protótipo ou até mesmo para a construção de um sistema comercial.

2 REVISÃO DE LITERATURA

O termo domótica deriva da junção da palavra em latim *domus*, que significa casa e a palavra robótica, ou seja, domótica torna possível controlar de forma automática uma residência. (FERREIRA, 2008).

O conceito de domótica segundo Bolzani:

É a ciência moderna de engenharia das instalações em sistemas prediais. A Domótica é uma ciência multidisciplinar que estuda a relação entre homem e casa. A imersão de pessoas em ambientes computacionalmente ativos revelou a necessidade do uso de técnicas mais sutis que gerenciassem a complexidade e o dinamismo das interações dos moderadores com o ambiente residencial saturado de diminutos dispositivos eletrônicos interligados a rede. (BOLZANI, 2010, p. 31).

A automação residencial, assim como a predial, é derivada da automação industrial, porém, com tecnologias adequadas para a realidade de uma residência, onde na maioria das vezes não há espaço suficiente para grandes centrais de controle e pesados sistemas de cabeamento (SENA, 2005).

2.1 História da Domótica

Segundo Moya e Tejedo (2010), a origem da domótica remete-se aos anos 70, quando surgiram os primeiros dispositivos de automação de edifícios, baseados na tecnologia X-10. O X-10 é um protocolo que permite controlar dispositivos em uma rede elétrica já existente, evitando a necessidade de novos cabeamentos.

É um sistema de fácil instalação, mas bastante instável, uma vez os componentes podem vir a falhar, ou até mesmo estragar devido à falta de energia ou até mesmo por descargas eletromagnéticas.

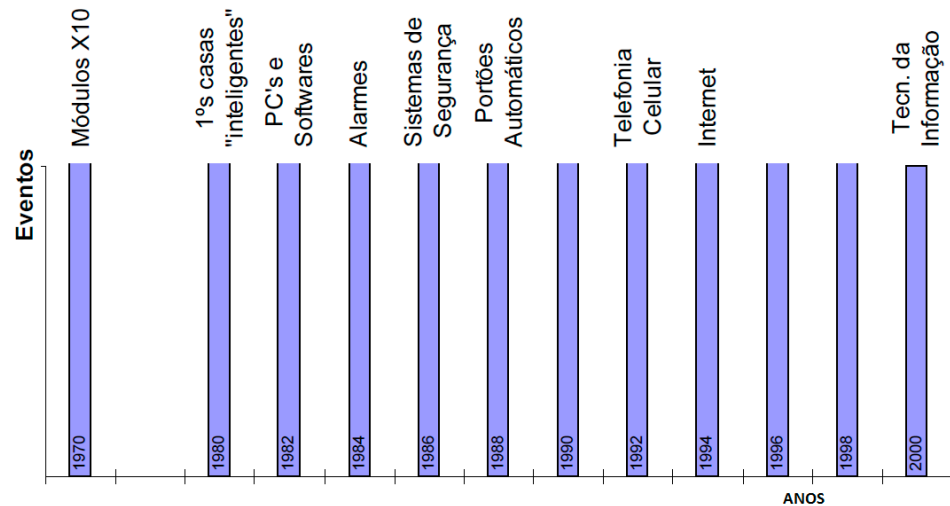
Nos anos 80, com o advento da computação pessoal, surgiram interfaces gráficas e operações muito mais simples que as já existentes, o que levou ao surgimento de novas possibilidades de automação residencial.

Mas foi no final da década de 90 que uma vasta gama de novidades surgiu, incorporadas com os telefones celulares e a web.

Atualmente a domótica utiliza esses novos recursos de forma integrada, como por exemplo, em sistemas de controle e monitoramento móvel, através de celulares, tablets, ou via web (OLIVEIRA, 2012).

A Figura 1 ilustra a evolução da domótica.

Figura 1 – História da Domótica



Fonte: AURESIDE, 2013

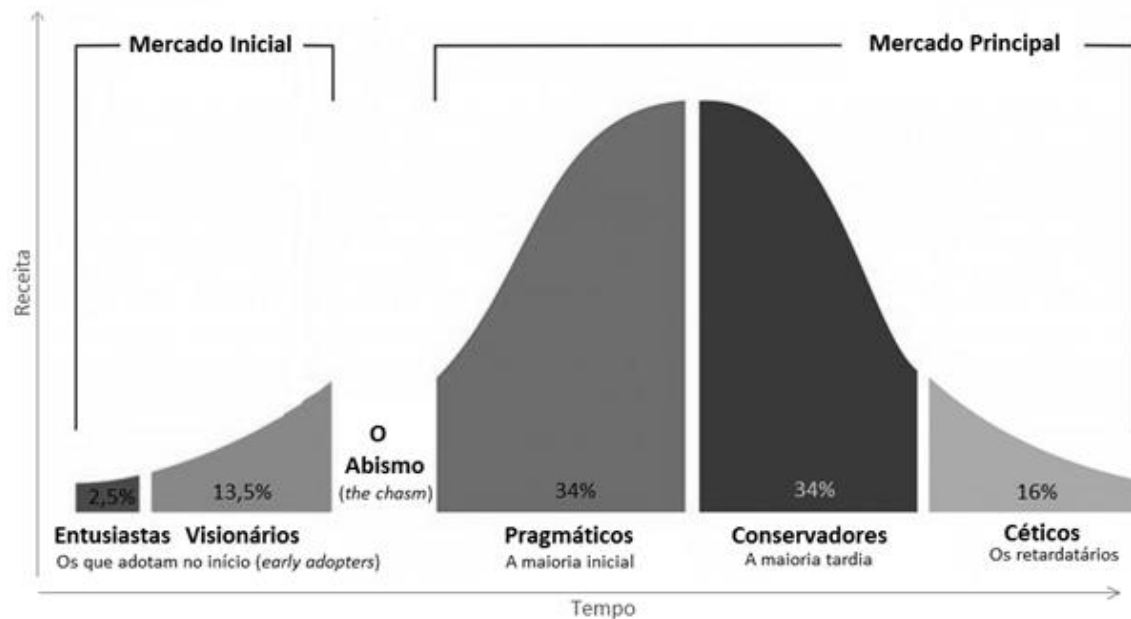
2.2 Desafios da Domótica

A próxima grande disputa das empresas de tecnologia será pela automação residencial, um mercado que deve movimentar 250 bilhões de dólares nos Estados Unidos nos próximos sete anos, e 1 trilhão de dólares em todo o mundo no mesmo período. Gigantes como Intel, Motorola, Microsoft e Cisco estão, neste momento, empenhados em criar a casa do futuro. Mas o maior problema é exatamente este: até agora, ninguém sabe o que ela precisa ter para agradar aos consumidores (EXAME 2005).

De acordo com pesquisas da Associação Brasileira de Automação Residencial, a busca por soluções de automação está em diferentes estágios nas regiões do Brasil. Em São Paulo, por exemplo, os interessados já não são mais os entusiastas ou visionários, mas também os pragmáticos e cidadãos comuns de classe média (TURUEL, 2008).

Para um produto ser mantido no mercado ele precisa de um número significativo de compradores, que seria atingido quando uma maioria inicial começasse a adquiri-lo, pois se isso não ocorre, o produto acaba caindo no esquecimento. O gráfico abaixo ilustra como é a curva de adoção de novas tecnologias.

Figura 2 - Ciclo de Adoção de Novos Produtos Pelos Consumidores



Fonte: GRANDO, 2013

A domótica já está deixando para trás esse abismo que existe entre consumidores visionários e pragmáticos.

No entanto, um dos grandes desafios que podem ser encontrados para o desenvolvimento da automação residencial, é a resistência dos próprios arquitetos e engenheiros eletricitas, seja por falta de conhecimento, por medo de novas tecnologias ou por necessidade da mudança da forma de trabalhar.

2.3 Características de sistemas de domótica

No mercado já existem diversas opções de sistemas de domótica, alguns mais simples que executam funções básicas como, por exemplo, controle de iluminação e controle de portas e portões, outros já permitem controlar até mesmo persianas, sons ambientes, temperatura da água da piscina e da banheira, entre outras diversas funcionalidades.

Entretanto, segundo a AURESIDE, algumas características são essenciais a qualquer sistema, como por exemplo:

- a) Capacidade de integrar todos os sistemas: os sistemas devem ser interligados através de uma rede, e permitir controle através de uma única interface;

- b) Atuação em condições variadas: o sistema deve ser capaz de atuar em condições adversas, como interrupções de energia, climas diversificados entre outros;
- c) Fácil relação com o usuário: os usuários muitas vezes não compreendem programações complexas, portanto, deve haver um sistema com interface intuitiva;
- d) Monitoramento: o monitoramento desse tipo de sistema é algo crítico, portanto devem ser realizadas auditorias com determinada frequência, e sempre observar relatórios de controle.

2.4 Benefícios proporcionados por sistemas de domótica

Os benefícios que podem ser observados imediatamente pelos moradores de uma casa automatizada são:

- a) Economia de energia: a energia é utilizada somente quando necessário, pois o controle da intensidade de iluminação, sensores de presença, controle da temperatura ambiente, elimina gastos desnecessários;
- b) Conforto: ajuste de temperatura de piscinas, filtros de ar, ar condicionado, entre outros equipamentos através de uma única interface;
- c) Conveniência: a temperatura do ambiente pode ser controlada mesmo antes da chegada dos moradores, lâmpadas e portões podem ser controlados de qualquer local;
- d) Acessibilidade: sistemas de automação com dispositivos touch pad e com reconhecimento de voz, proporcionam a portadores de necessidades especiais, a possibilidade de controlar o ambiente que estão, seja ascendendo uma lâmpada, controlando luminárias, televisores, portas, entre outros, devolvendo ao indivíduo sua independência;
- e) Segurança: a possibilidade de controle de luminosidade, ar condicionado, televisores e outros dispositivos, podem fazer que a residência pareça sempre ocupada.

Câmeras de monitoramento podem ser acessadas de qualquer local, o que permite que a casa esteja sempre monitorada, um exemplo citado por LAGUÁRDIA, foi o caso do empresário Ronan Soares. O empresário estava

na cidade de Colônia, na Alemanha, ao verificar a câmera de monitoramento de sua residência, percebeu que havia ladrões invadindo o local, rapidamente avisou a esposa que estava no Brasil, que praticamente no mesmo instante acionou a polícia.

2.5 Mercado da domótica

A automação residencial, ou domótica, como também é conhecida, está em uma fase de muito crescimento no Brasil, nos últimos quatro anos o serviço cresceu 300%, e já existem 25 empresas do setor no país (AURESIDE).

Dados da Associação Brasileira de Automação Residencial revelam que 300 mil residências do Brasil possuem algum tipo de automação, e este número pode vir a crescer muito mais, pois de acordo com pesquisa realizada pela associação, 78% dos brasileiros possuem interesse nesse serviço, um número maior que a média mundial, 66%.

Segundo a AURESIDE em 2012 o segmento faturou R\$ 4 bilhões e pode crescer mais de 30% até o fim desse ano.

2.6 Sistemas Comercializados

Realizou-se uma pesquisa de sistemas de domótica que estão disponíveis para comercialização no Brasil, observou-se que em todos o principal objeto de automação é sistema de iluminação, como se pode verificar nos sub tópicos a seguir.

2.6.1 IHouse

É um sistema que promete realizar a automação residencial sem obras ou mudanças na infraestrutura da residência.

Proporciona o controle da iluminação, com graduação entre 0 e 100%, comando de ar condicionado, persianas e ajuste de fluxo e temperatura da água para banho.

Não proporciona acesso fora da residência, uma vez que o controle e programação se dão somente através do painel do Wallpad (um controle que se comunica através de sinal wireless com os módulos instalados) abaixo ilustrado.

Figura 3 – Walpad IHouse



Fonte: IHOUSE, 2013

2.6.2 GDS Automação

Sistema de automação residencial que proporciona o controle de iluminação, som, *home theater*, ar condicionado e sistemas de irrigação de jardins, através de tablets ou celulares.

O projeto é desenvolvido de acordo com o desejo e a necessidade do cliente, sendo a interface personalizável para cada projeto. Abaixo a ilustração da tela de controle de uma sala.

Figura 4 – Controle de sala GDS Automação



Fonte: GDS, 2013

2.6.3 Simplifies

Sistema desenvolvido pelo grupo QualiHouse Automação Predial, composto por módulos de automação, servidor Web, sensores, atuadores, câmeras e ativos de rede, que podem ser switches, roteadores, etc.

É um sistema que permite ao cliente automatizar sua residência aos poucos, uma vez que são instalados módulos independentes em cada local. Esses módulos realizam a interface entre o sistema executado no servidor e os sensores e atuadores instalados.

O usuário tem possibilidade de ligar/desligar lâmpadas, abrir portas, monitorar sensores, agendar eventos, visualizar câmeras, controlar música ambiente entre outras funcionalidades, através de qualquer dispositivo com acesso à Internet.

3 COMPONENTES FÍSICOS DO PROTÓTIPO

Para a construção protótipo de automação residencial, que será descrito no capítulo cinco, foram utilizados diversos componentes físicos, cujo principal deles é o Arduino.

O módulo *Ethernet* também possui função significativa, uma vez que será ele que irá prover a conexão do Arduino com a Internet.

Para melhor compreender a integração de todos esses componentes, é necessário saber como se dá o funcionamento do protótipo.

Primeiramente, o usuário enviará o comando que deseja executar, como ascender ou apagar uma lâmpada, abrir o portão ou verificar as imagens da câmera de segurança, através da página desenvolvida em PHP. Essas informações serão recebidas pelo roteador, se a opção do usuário foi verificar a câmera, será direcionado diretamente para a página com as imagens, se a opção foi ascender/apagar ou abrir/fechar o portão, o comando será enviado para o Arduino, que por sua vez irá processar as informações e realizar a interface com as lâmpadas e o servomotor.

A figura a seguir ilustra a interação entre esses componentes do sistema.

Figura 5 – Interação entre componentes do sistema



Fonte: Adaptado pela autora

3.1 Plataforma Arduino

Para trabalhar como o “cérebro” do projeto, foi escolhida a plataforma de prototipagem eletrônica Arduino, devido ao fato de ser uma placa robusta, de baixo custo e com linguagem de programação de fácil aprendizagem.

A seguir, será apresentado sua história e como consiste seu funcionamento.

3.1.1 O que é o Arduino?

Arduino é uma plataforma de prototipagem eletrônica de código aberto, ou seja, qualquer pessoa no mundo pode ter acesso à seu código, sem pagar por isso. É baseado em uma simples placa de entrada e saída e um ambiente de desenvolvimento.

O Arduino é formado por um micro controlador Atmel AVR, com entradas e saídas analógicas e digitais, interligados em uma placa de prototipagem (OLIVEIRA, 2012).

A plataforma possui seu próprio ambiente de desenvolvimento com linguagem de programação baseada em C/C++.

3.1.2 História do Arduino

O Arduino é um projeto iniciado em 2005, na cidade de Ivrea, na Itália, pelo professor Massimo Banzi. Este professor queria ensinar eletrônica e programação de computadores para seus alunos do curso de design, porém encontrava muita dificuldade, uma vez que não eram alunos da área de programação ou eletrônica, e devido à inexistência de placas robustas e de baixo custo no mercado.

Pensando nestes problemas, Massimo Banzi e o engenheiro eletrônico David Cuartielles decidiram projetar sua própria placa, juntamente com a ajuda de um aluno de Massimo, David Mellis, que ficou responsável por desenvolver a linguagem de programação da plataforma.

Gianluca Martino foi o responsável por criar o protótipo comercial para fabricação em grande escala. A princípio foram fabricadas somente duzentas placas, vendidas a escolas com o lucro de cerca de um euro por cada. Meses depois, a

empresa americana *Sparkfun*, decidiu comercializar o projeto, começando com 20 placas, em 2010 estima-se que foram vendidas 40.000.

Assim iniciou a história do Arduino, que atualmente é utilizado no mundo inteiro, nos mais diversos tipos de projetos.

3.1.3 Ambiente de Desenvolvimento

O ambiente de desenvolvimento consiste em um software gratuito, onde será escrito a sequência de instruções que serão interpretadas pelo Arduino.

Ele se conecta ao hardware para realizar a comunicação e carregar o código desenvolvido.

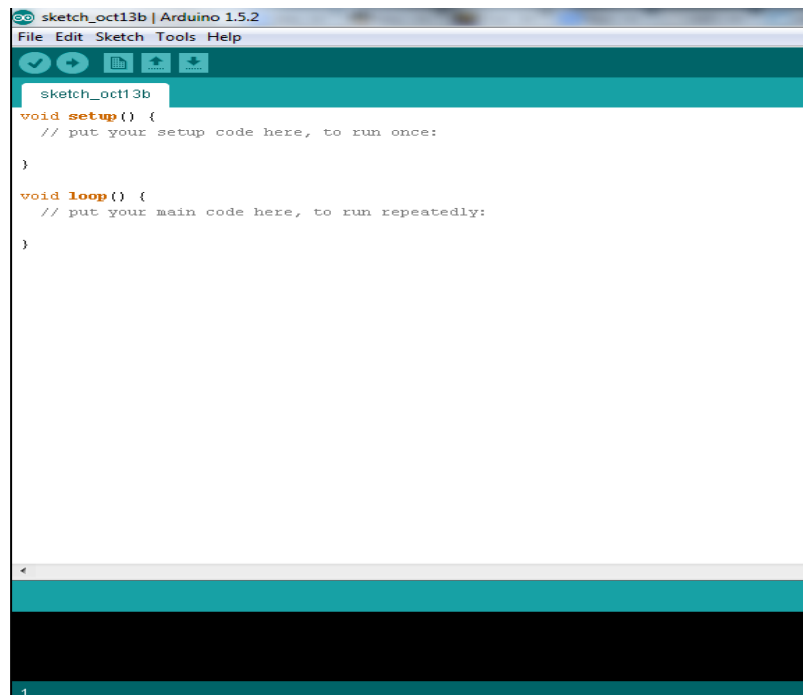
Os códigos escritos neste ambiente de desenvolvimento são chamados de *Sketches*, que são salvos a extensão *.INO*.

Quando se inicia a IDE (*Integrated Development Environment*) encontra-se a área para escrever o software, a barra de ferramentas, o console de textos, que exibe uma lista completa de erros no código e o resultado das instruções enviadas ao Arduino, e os seguintes botões:

- a) *Verify*: verifica prováveis erros no código;
- b) *Upload*: compila o código e carrega para a placa;
- c) *New*: cria um novo esboço;
- d) *Open*: apresenta um menu de vários códigos prontos;
- e) *Save*: salva o sketch;

Há também alguns comandos adicionais oferecidos para facilitar o desenvolvimento, como por exemplo: *Copy for fórum*, encontrado dentro do menu editar, que torna possível copiar o código para postar em fóruns, *Copy as HTML*, oferece opção de copiar como HTML para inserir em páginas Web, *Import Library*, que adiciona uma biblioteca ao projeto, *Examples*, onde encontra-se diversos exemplos de código prontos, entre outras opções que tornam essa ferramenta muito simples de trabalhar.

Figura 6 – Ambiente de desenvolvimento do Arduino



Fonte: Adaptado pela autora

3.1.4 Bibliotecas

As bibliotecas são conjuntos de códigos disponibilizados pela desenvolvedora do projeto Arduino, que têm por objetivo facilitar a comunicação com os componentes acoplados à placa.

Existem diversas bibliotecas disponíveis, algumas são internas, como por exemplo: *Servo* e *Ethernet*, que foram utilizadas no desenvolvimento do projeto de automação residencial, outras são disponibilizadas para *download* e podem ser instaladas muito facilmente.

Abaixo exemplo de como adiciona-se uma biblioteca ao código:

```
#include <Ethernet.h>
#include <Servo.h>
```

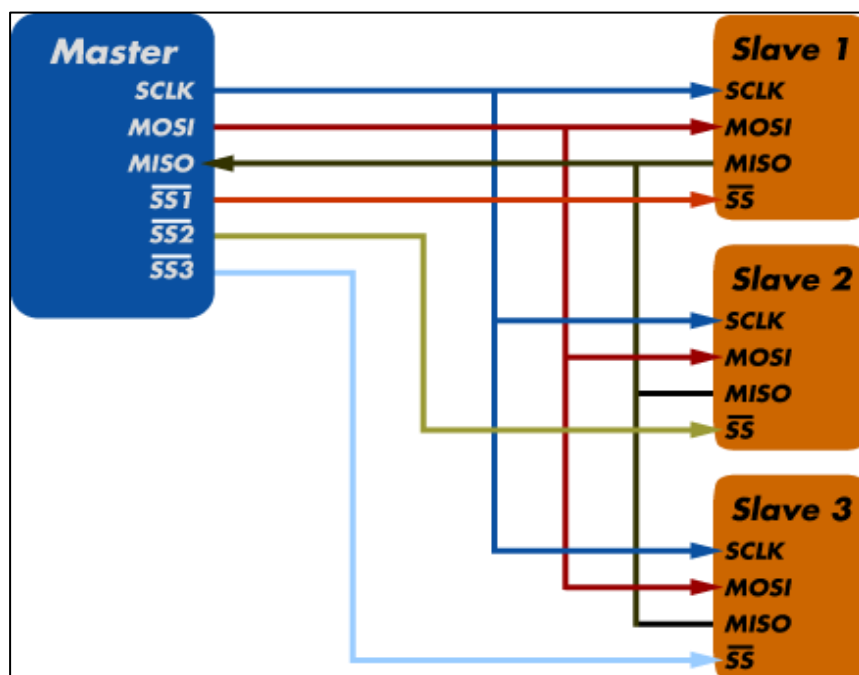
3.1.4.1 Biblioteca SPI

A biblioteca SPI permite a comunicação entre dispositivos de SPI (Serial Peripheral Interface) e o Arduino.

O SPI é um protocolo de dados seriais síncronos, geralmente utilizado em micro controladores para comunicação entre um ou mais periféricos. Na comunicação SPI sempre há um periférico mestre, neste caso, o Arduino é o mestre e os demais periféricos, como o módulo Ethernet, são escravos. Nessa comunicação há quatro conexões, que são:

- a) MISO (Master In Slave Out) - Dados do Slave para Master;
- b) MOSI (Master Out Slave In) - Dados do Master para Slave;
- c) SCK (Serial Clock) - Clock de sincronização para transmissão de dados entre o Master e Slave;
- d) Hardware SS (Slave Select) - Seleciona qual Slave receberá os dados.

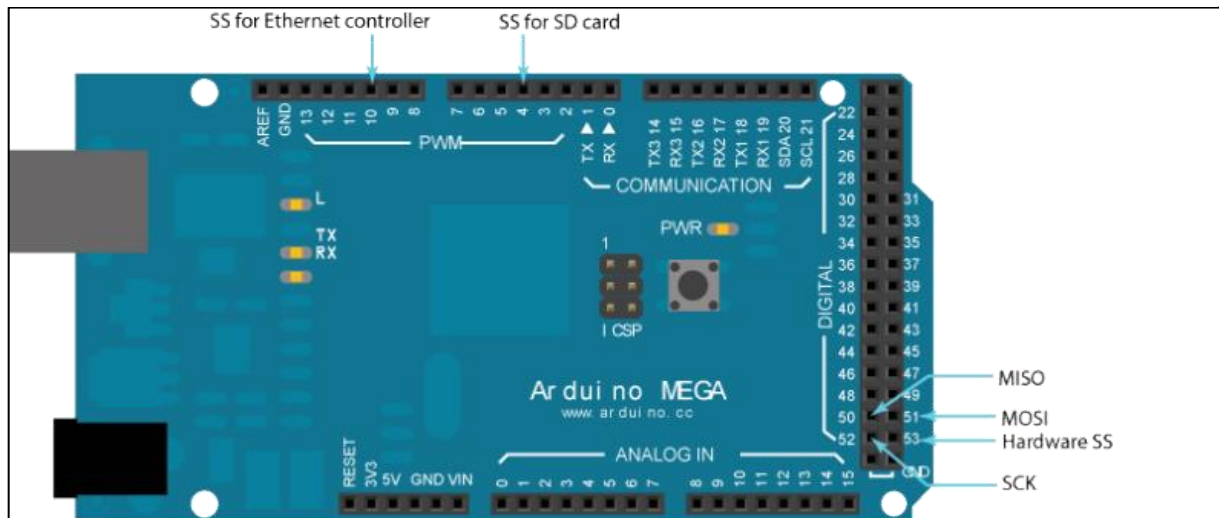
Figura 7 – Comunicação SPI



Fonte: COMUNICAÇÃO SPI, 2013

O Arduino Mega 2560 se comunica com o módulo *Ethernet* através do barramento SPI, utilizando os pinos 50, 51 e 52, ilustrados abaixo:

Figura 8 – Pinos utilizados para comunicação SPI



Fonte: ARDUINO, 2013

3.1.4.2 Biblioteca Ethernet

Para tornar possível o Arduino se conectar à internet é necessário acoplar a ele um módulo *ethernet*. E, para uma fácil comunicação entre estes dois componentes utiliza-se a biblioteca *ethernet*.

Para iniciar a biblioteca *Ethernet* é chamado o método `Ethernet.begin`, onde deve-se passar o parâmetros *mac*, *IP*, *gateway* e *subnet*, como exemplificado no trecho de código destacado abaixo:

Figura 9 - Trecho de código chamada da biblioteca Ethernet

```
void setup() {
  Ethernet.begin(mac, ip, gateway, subnet);
  server.begin();
  pinMode(A0, OUTPUT);
  pinMode(A1, OUTPUT);
  pinMode(A2, OUTPUT);
  pinMode(A3, OUTPUT);
  pinMode(A4, OUTPUT);
  pinMode(A5, OUTPUT);
}
```

Fonte: Adaptado pela autora

3.1.4.3 Biblioteca Servo

Permite ao Arduino controlar servo motores. Na maioria das placas suporta até 12 motores, já o Arduino Mega 2560 suporta 48 (ARDUINO, 2013).

No código abaixo, observa-se as funções que foram utilizadas no projeto, como por exemplo, a função servo, quando se cria o objeto para controlar o servo motor;

Figura 10 – Trecho de código utilizando biblioteca Servo

```
#include <SPI.h>
#include <Ethernet.h>
#include <Servo.h>
Servo servol; // Criação de um objeto que permitira controlar o servo
int posicao = 0; // inicialização da variavel posição a 0
int contagem = 0;
```

Fonte: Adaptado pela autora

Attach: quando atribui-se o pino que o servomotor será conectado, conforme visualiza-se na Figura 11.

Figura 11 – Pino onde o servomotor é conectado

```
void setup() {
  Ethernet.begin(mac, ip, gateway, subnet);
  server.begin();
  pinMode(A0, OUTPUT);
  pinMode(A1, OUTPUT);
  pinMode(A2, OUTPUT);
  pinMode(A3, OUTPUT);
  pinMode(A4, OUTPUT);
  servol.attach(13);
}
```

Fonte: Adaptado pela autora

O comando *Write*, responsável por informar a posição para onde o servo deslocará.

Figura 12 – Utilização do comando Write

```
for(posicao = 180; posicao>=90; posicao-=1)    // Servo gira da posição 180 para a posição 0 de 1 em 1 grau
{
  servo.write(posicao);                      // "Escreve" no pino 13 a posição para onde o servo se vai deslocar
  delay(2);                                  // Espera de 2 milisegundos para que o servo possa chegar á sua posição
}
```

Fonte: Adaptado pela autora

3.1.5 Arduino Mega

O projeto desenvolvido utilizará o modelo Arduino Mega 2560, devido ao fato de ser uma placa recente, que possui compatibilidade com a maioria dos módulos existentes, e por possuir uma quantidade maior de pinos de entrada e saída se comparado à versão mais popular, Arduino Uno, que possui 14, enquanto o Mega possui 54.

A versão Mega consiste em uma placa micro controlada, onde o processador é o micro controlador ATmega 2560.

Figura 13 – Arduino Mega



Fonte: ARDUINO, 2013

3.1.5.1 Características

O Arduino Mega possui 54 pinos digitais, dos quais 15 podem ser utilizados como saídas PWM (*Pulse Width Modulation*), 16 entradas analógicas, 4 portas seriais de hardware, uma conexão USB, um conector de alimentação, um cabeçalho ICSP, e um botão de *reset* (ARDUINO, 2013).

Abaixo o Quadro 1 evidencia melhor suas características.

Quadro 1 - Características do Arduino Mega

Micro controlador	Atmega 2560
Tensão de funcionamento	5V
Tensão de entrada (recomendado)	7 a 12V
Tensão de entrada (máxima)	6 a 20V
Pinos de entrada e saída digital	54 (dos quais 14 podem ser saídas PWM)
Pinos de entradas analógicas	16
Valor máximo de corrente fornecida por pino	40mA
Valor de corrente para pino 3,3V	50mA
Memória flash	256KB
SRAM	8KB
EEPROM	4KB
Velocidade de clock	16MHz

Fonte: ARDUINO 2013

3.1.5.2 Função dos pinos

Todos os 54 pinos podem ser utilizados como entrada ou saída, através das funções `pinMode`, `digitalWrite` e `digitalRead`, no trecho de código abaixo a função `pinMode` aciona os pinos analógicos A0, A1, A2, A3, A4, A5, A6, A7 e A8 para trabalharem como entrada e saída de dados, realizando o controle das lâmpadas do protótipo.

Figura 14 – Pinos Arduino

```
void setup() {
  Ethernet.begin(mac, ip, gateway, subnet);
  server.begin();
  pinMode(A0,OUTPUT);
  pinMode(A1,OUTPUT);
  pinMode(A2,OUTPUT);
  pinMode(A3,OUTPUT);
  pinMode(A4,OUTPUT);
  pinMode(A5,OUTPUT);
  pinMode(A6,OUTPUT);
  pinMode(A7,OUTPUT);
  pinMode(A8,OUTPUT);
  servol.attach(13);
}
```

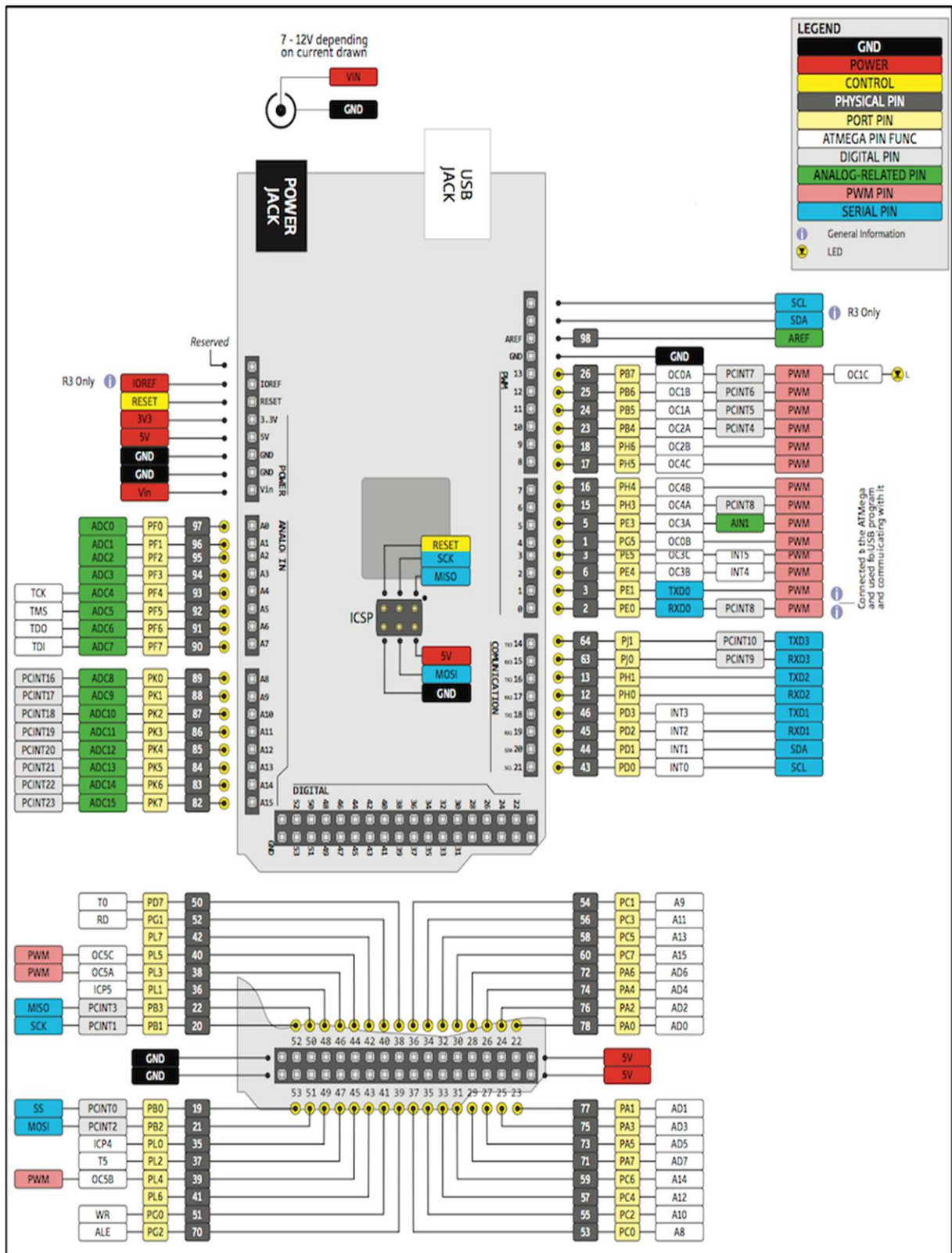
Fonte: Adaptado pela autora

Os 15 pinos que podem ser utilizados como saídas PWM, operam por modulação de largura de pulso, ou seja, quando uma onda quadrada alterna seu estado em nível lógico alto ou baixo (ou no sistema binário 0 e 1).

O pino 13 é um dos que possuem esta função, e será utilizado no projeto para controlar o servo motor.

Alguns pinos possuem funções especiais, como por exemplo, TX e RX que transmitem e recebem dados seriais. A imagem a seguir ilustra onde estão localizados os pinos e quais as suas funções.

Figura 15 – Pinos do Arduino Mega 2560



Fonte: PIGHIXX, 2013

3.2 Módulo Ethernet W5100

O módulo ethernet W5100 é compatível com o Uno Arduino e Mega, ele permite que o Arduino acesse a Internet como servidor ou cliente, através de um cabo RJ45.

Sua conexão se dá pelo barramento SPI (Serial Peripheral Interface), através das saídas digitais 11, 12 e 13.

Ele é baseado no chip ethernet W5100 Wiznet, que é responsável por fornecer pilha TCP ou UDP.

O módulo possui uma série de LEDs de informação, abaixo o significado de cada um deles:

Quadro 2 - LEDs de Informação

LED	Significado
PWR	Indica que o módulo está ligado
LINK	Indica a presença de um link de rede e pisca quando o módulo transmite ou recebe dados
FDX	Indica que a conexão de rede é full duplex
SPD	Indica a presença de uma conexão de rede de 100 Mb/s (oposto a 10 Mb/s)
RX	Pisca quando o módulo recebe dados
TX	Pisca quando o módulo envia dados
COL	Pisca quando colisões de rede são detectadas

Fonte: ARDUINO, 2013

Caso seja necessário o armazenamento de alguma informação, possui também espaço para inserir micro SD. Na Figura 16, visualiza-se como é o módulo ethernet 5100.

Figura 16 – Módulo Ethernet



Fonte: Fotos da autora

O W5100 não possui um MAC address predefinido, portanto, foi atribuído a ele, através da função *Ethernet.begin* o MAC address DE-AD-BE-EF-FE-ED, juntamente de um IP fixo 192.168.1.102, gateway padrão 192.168.1.1 e máscara de rede 255.255.255.0, conforme o código abaixo:

Figura 17 – Configurações do Arduino

```
//Configurações do Ethernet Shield
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };
byte ip[] = { 192,168,1, 102 }; // ip que o arduino assumirá
byte gateway[] = { 192,168,1,1 }; // ip do roteador
byte subnet[] = { 255, 255, 255, 0 };
```

Fonte: Adaptado pela autora

3.3 Servo motor

Servo motor é um dispositivo eletromecânico de posição controlada. Seu eixo pode ser posicionado em vários ângulos entre 0° e 180°.

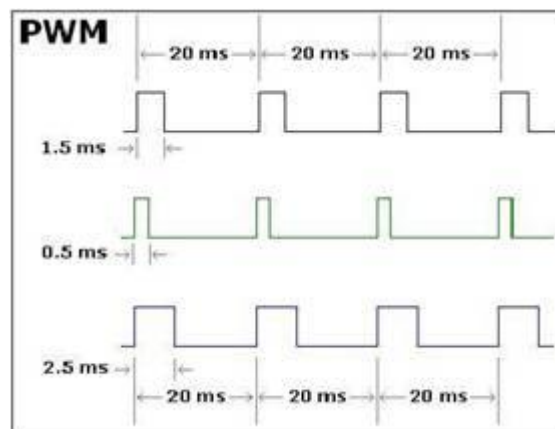
O Arduino possui uma biblioteca própria para trabalhar com servo motores. Na maioria das placas são suportados até 12 servos, já no Arduino Mega, possibilita a conexão de 48.

Ele é composto basicamente por quatro partes, que são:

- a) Circuito de controle: recebe os sinais de controle para determinar o ângulo que o servo motor será posicionado.
- b) Potenciômetro: monitora a posição do servo motor;
- c) Motor: movimenta as engrenagens.
- d) Engrenagens: Movimentam o potenciômetro e reduzem a rotação do motor.

O ângulo do eixo do servo motor será determinado pela duração da largura do pulso recebido, por exemplo, se for recebido um pulso com duração de 1.5 milissegundo, ele será girado até o meio, se reduzir até 1 milissegundo irá para um lado, se aumentar até 2 milissegundos, será girado até o outro lado.

Figura 18 – Princípio de funcionamento de um servo motor



Fonte: JÚNIOR; SIQUEIRA

3.4 Câmera IP

A câmera IP é um dispositivo que pode ser controlado e acessado através de um endereço IP. Ela possui um servidor interno, capaz de armazenar as imagens filmadas.

Através de qualquer navegador Web é possível acessar as imagens e áudio, que são transmitidos online.

Para seu funcionamento é necessário somente um roteador, que irá fornecer o endereço IP para acesso à câmera.

Neste projeto será utilizada a câmera Tenvis JPT3815W, para configurá-la, basta instalar seu driver no computador que utilizará para configurá-la - nos demais computadores que acessarão suas imagens não há necessidade – em seguida, basta conectá-la ao roteador e atribuir um endereço IP fixo.

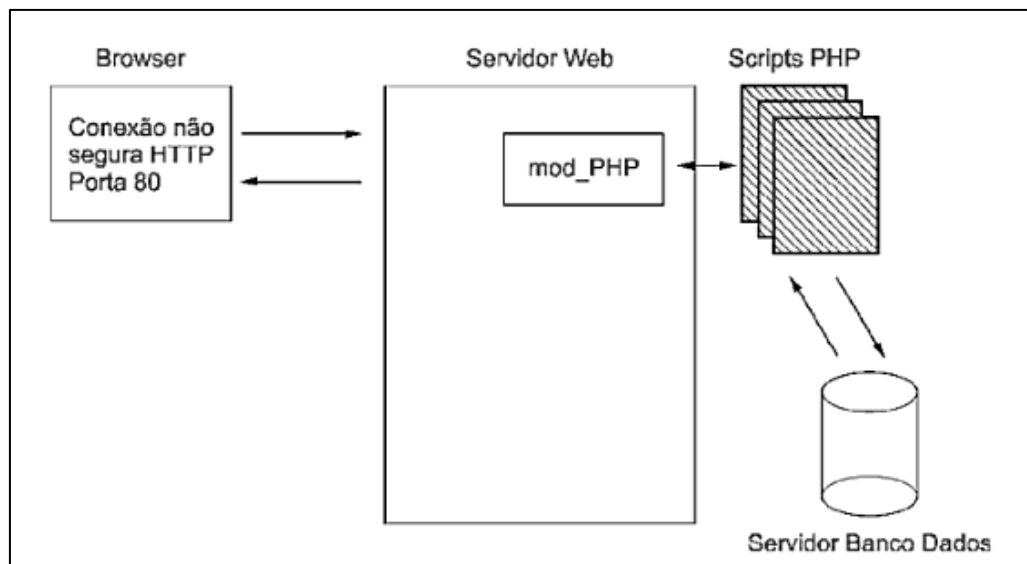
4 TECNOLOGIAS PARA DESENVOLVIMENTO DO SISTEMA WEB

Para o desenvolvimento e disponibilização da página Web do protótipo, foram utilizadas as seguintes ferramentas: servidor Web Apache, servidor de banco de dados MySQL e linguagem de programação PHP (*Hypertext Preprocessor*).

As três operam de forma integrada, da seguinte maneira: quando o usuário solicitar uma página web, o servidor Apache solicita ao pré-processador PHP que execute a solicitação, se na página houver comandos que abrem conexão com banco de dados, o PHP irá interpretar a chamada SQL e extrair as informações do banco. Em seguida, o MySQL enviará a resposta e resultados ao interpretador PHP e, por fim, o Apache enviará a resposta colhida pelo PHP de volta ao navegador, no formato HTML. Enfim a página será apresentada ao usuário. Dessa forma, é possível interagir com o banco de dados e as aplicações existentes no servidor de forma segura, pois o código não está sendo exposto para o cliente.

A Figura 19 ilustra esse funcionamento descrito.

Figura 19 – Funcionamento PHP, servidor Web e banco de dados



Fonte: adaptado pela autora

4.1 PHP

PHP (*Hypertext Preprocessor*) é uma linguagem de programação voltada para o desenvolvimento Web.

Segundo CHOI e outros (2001), em termos técnicos, PHP é uma linguagem de *script* da Web, rodando em um servidor multiplataforma (ou seja, em quaisquer sistemas operacionais).

Os sistemas desenvolvidos combinam PHP e HTML, sendo que o primeiro é para controlar e o último para formatar (CHOI, 2001).

4.1.1 História do PHP

A linguagem de programação PHP surgiu no ano de 1994. Foi desenvolvida por Rasmus Lerdof na linguagem de programação C.

Inicialmente nomeada de Personal Home Page Tools, foi criada para substituir o conjunto de scripts em Perl, que utilizava no desenvolvimento de sua página pessoal.

Ao longo tempo, várias funcionalidades começaram a ser adicionadas, e em 1997 foi lançado a PHP/FI, com a ferramenta *Forms Interpreter*, que proporcionava interpretar comandos SQL.

A PHP se expandiu de forma muito rápida, em 1998, uma pesquisa da Netcraft, indicou que cerca de 1% de todos os domínios da Internet já utilizavam a linguagem.

Nos anos seguintes foram lançadas novas versões, a primeira foi a PHP 3.0, lançada com um novo nome, simplesmente PHP, com o significado tornando um acrônimo recursivo *Hypertext Preprocessor*.

A versão proporcionava interface com diversos bancos de dados, protocolos e APIs, mas a principal delas foi a inclusão de suporte à programação orientada a objetos.

Em maio de 2000 a versão 4.0 da linguagem foi oficialmente lançada, com algumas melhorias, como suporte à maioria dos servidores web, sessões HTTP, maneiras mais seguras de manipulação de dados e maior desempenho.

A última versão foi lançada em julho de 2004, quando foi introduzido um novo modelo de orientação a objeto, o tratamento de objetos foi reescrito, trazendo maior facilidade no desenvolvimento.

Atualmente, atualizações de correção de problemas, são lançadas frequentemente, a última delas foi a 5.5.0, lançada em 20 de junho desse ano.

4.1.2. Vantagens da PHP

Existem diversas vantagens ao utilizar a linguagem de programação PHP, as principais que se pode citar são:

- a) Curva de aprendizado curta - permite que programadores que nunca tiveram contato com linguagem compreendam rapidamente seu funcionamento;
- b) Maior comunidade da web – linguagem mais popular, possui diversos fóruns de discussão, onde programadores frequentemente compartilham seus conhecimentos;
- c) Utilização gratuita;
- d) Multiplataforma;
- e) Possibilita utilização dos maiores e mais utilizados banco de dados, como Adabas, MySQL, Oracle, entre outros;
- f) Interface de desenvolvimento simples.

4.2 Apache

Para tornar possível o acesso à página Web desenvolvida em PHP (*Hypertext Preprocessor*), será utilizado o servidor Web Apache.

O Apache será responsável por receber as requisições enviadas pelo sistema Web, e por sua vez responder às solicitações com códigos em HTML que serão interpretados pelo navegador.

“Segundo levantamento feito pela Netcraft (<http://netcraft.com/survey>), em junho de 2009, cerca de 50% dos hospedeiros de páginas Web em operação empregam servidor Apache. O segundo posto é ocupado por servidores Microsoft (Microsoft-IIS e Microsoft-PWS) com 25% dos sítios, e os demais utilizam soluções tipo

lplanet-Enterprise, Netscape-Enterprise, dentre outros” (BRUNO; ESTROZI; NETO, 2010, p. 7).

A tabela a seguir demonstra essa liderança de mercado do Apache em números.

Tabela 1 – Números servidores Web

Servidor Web	Julho/2007	Porcentagem
Apache	66.144.734	52,65%
Microsoft	41.257.913	32,84%
Google	5.465.538	4,35%
Sol	2.245.493	1,79%
Lighttpd	1.471.779	1,17%
Zeus	463.449	0,37%

Fonte: NETCRAFT, 2013

Os principais fatores que impulsionaram o Apache a deter tal porcentagem de mercado foram sua rapidez, sua simplicidade de configuração, seu funcionamento superior ao dos concorrentes, entre diversos outros.

4.3 MySQL

O banco de dados utilizado no desenvolvimento da interface Web do protótipo, foi o MySQL, devido ao fato de ser um sistema de gerenciamento de banco de dados gratuito e de interface simples.

O MySQL foi desenvolvido pela empresa sueca de consultoria TcX em 1980. Eles estavam precisando de um banco de dados que fosse extremamente rápido e flexível, porém não estavam encontrando no mercado, portanto, criaram o MySQL, que é vagamente baseado no MSQL (MASLAKOWSKI, 2000).

Esse sistema de gerenciamento de banco de dados é amplamente utilizado em todo o mundo, e em grandes organizações, como NASA, HP, Bradesco, Sony, dentre outras.

O MySQL é um banco de dados do tipo relacional, ou seja, é composto por tabelas e colunas que se relacionam entre si, relacionamento que é baseado em uma chave contida em alguma coluna.

A vantagem de utilizar um banco de dados relacional é por ele ser muito intuitivo, uma vez que simula a maneira com as pessoas pensam. As pessoas tendem a agrupar objetos semelhantes e separar objetos complexos em objetos mais simples, e é essa a natureza desse tipo de banco de dados (MASLAKOWSKI, 2000).

4.3.1 PHPMyadmin

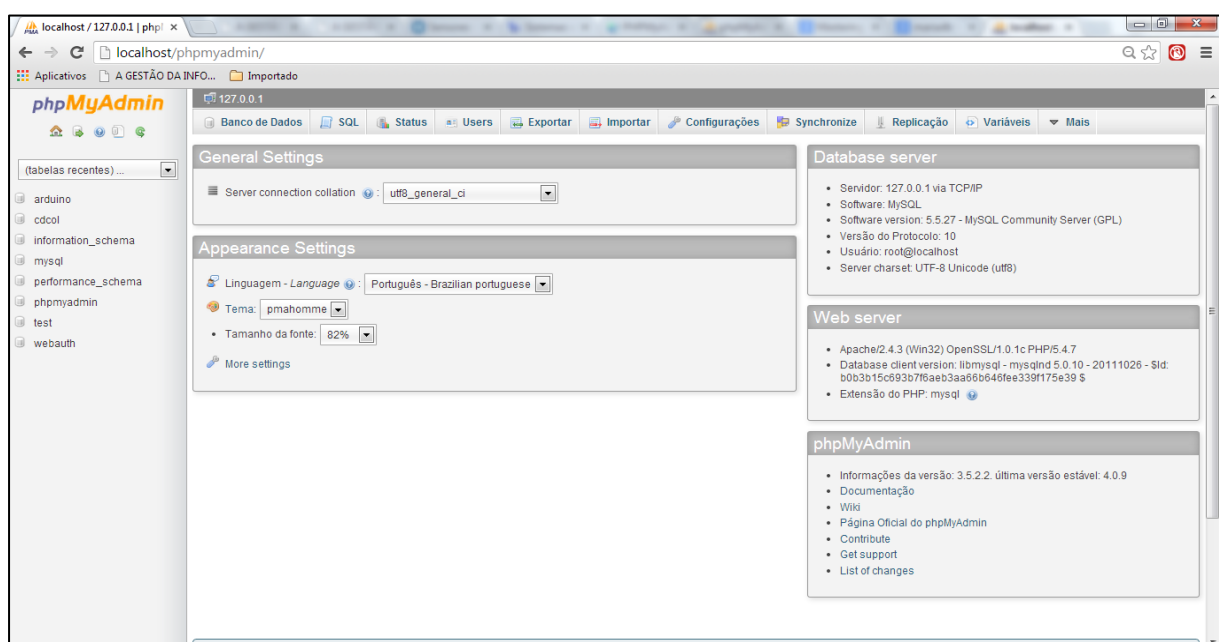
O PHPMyadmin é uma ferramenta de software livre, desenvolvida em PHP, para realizar a administração do banco de dados MySQL pela Web (DELISLE, 2012).

Segundo o site oficial da desenvolvedora da ferramenta, o PHPMyadmin suporta uma vasta gama de operações em MySQL e MariaDB, sendo que as mais utilizadas são o gerenciamento de tabelas, colunas, índices, usuários, permissões, etc.

Possui uma interface muito simples de utilizar, onde é possível realizar o gerenciamento das tabelas, controle de permissão de usuários, verificação de estatísticas de utilização, entre outros.

A figura 20 ilustra a página inicial da ferramenta.

Figura 20 – Página inicial do PHPMyadmin



Fonte: Adaptado pela autora

Existem diversas vantagens em utilizar o PHPMyadmin, como:

- a) Possui interface Web intuitiva;
- b) Suporte para a maioria dos recursos do MySQL;
- c) Proporciona importação de dados de CSV ou SQL;
- d) Proporciona exportação de dados para diversos formatos, como CSV, SQL, XML, PDF, ISO e outros;
- e) Administração de diversos servidores, entre outras vantagens.

5 DESENVOLVIMENTO DO PROTÓTIPO

Para conclusão desse trabalho, foi implementado um projeto prático, para melhor visualização e aplicação dos conceitos já citados anteriormente. Portanto foi desenvolvido um protótipo de um sistema de automação residencial, nomeado como Domus Automação Residencial.

Tem-se por objetivo futuro torna-lo um sistema comercial, mas a princípio será utilizado somente para testes.

O objetivo foi desenvolver um sistema capaz de controlar uma maquete de dimensões 80cm x 65cm, através de qualquer dispositivo com acesso à Internet.

O sistema possui uma interface simples de operar, sendo que qualquer pessoa é capaz de utilizá-lo, sem a necessidade de um treinamento prévio.

Integra todas as ferramentas de desenvolvimento e componentes físicos citados ao decorrer desse trabalho.

Mas antes de começar a descrever o protótipo, é necessário compreender o que mudará em relação a uma casa que não possui automação.

5.1 O que muda ao Automatizar uma residência?

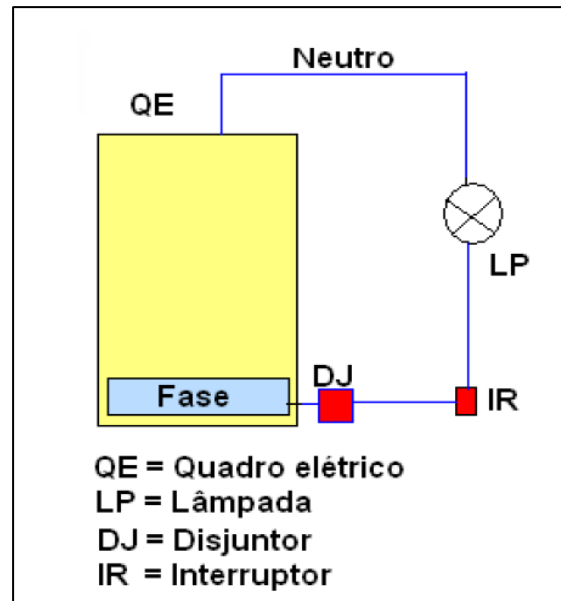
A principal mudança em uma casa automatizada, é a presença de um quadro de automação, que receberá a ligação de todos os aparelhos controlados pelo sistema de automação residencial.

É muito mais simples realizar a automação de uma residência já planejada para esse tipo de sistema, no entanto é possível fazer algumas adaptações em casas já prontas, cujos proprietários desejam automatizá-las.

Serão necessários alguns ajustes no cabeamento da residência, pois eles deverão interligar componentes automatizados ao QA (quadro de automação).

A Figura 21 demonstra como é a ligação de uma lâmpada sem automação.

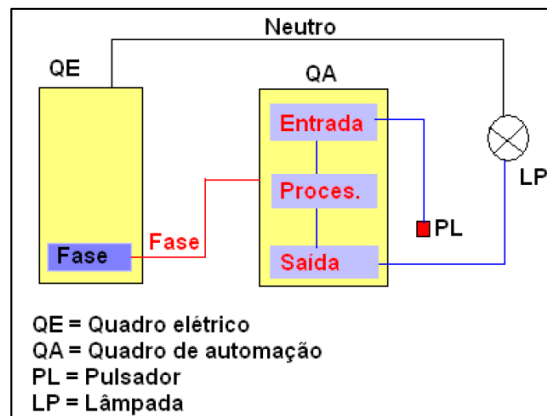
Figura 21 – Ligação de lâmpada sem automação



Fonte: TURUEL, 2008

A Figura 22 ilustra mostra quais as mudanças serão necessárias no esquema elétrico para a ligação de uma lâmpada em um sistema automatizado.

Figura 22 – Ligação de uma lâmpada ao quadro elétrico com quadro de automação



Fonte: TURUEL, 2008

Perceba que no lugar do interruptor comum foi inserido um pulsador, que na verdade é o sinal que é enviado ao Arduino através da página Web.

Não somente as lâmpadas, mas todos os aparelhos que antes eram ligados diretamente a um quadro elétrico, no ambiente automatizado, passarão primeiramente pelo quadro de automação, que neste caso é o Arduino.

5.2 Desenvolvimento da maquete

Primeiramente foi projetado no AutoCad a planta de uma residência de sete cômodos, sendo dois quartos, uma sala, cozinha, copa e dois banheiros.

A Figura 23 ilustra a planta da maquete.

Figura 23 – Planta da maquete



Fonte: Adaptado pela autora

Nela foram inseridos LEDs em todos os cômodos e jardim, que simulam as lâmpadas e um servo motor, que simula o motor de um portão, todos controlados pela Web, por intermédio do Arduino, e uma câmera de monitoramento, que disponibilizará imagens da residência em tempo real e também enviará e-mails caso seja detectado algum movimento em suas proximidades.

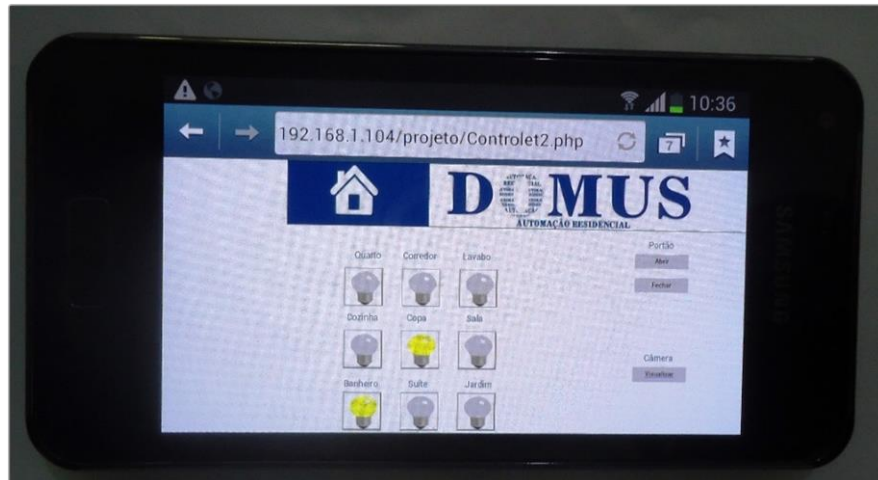
5.3 Desenvolvimento da página Web

Todo o sistema Web foi desenvolvido na linguagem de programação PHP.

Tornando assim possível o acesso através de qualquer dispositivo com acesso à Internet.

A Figura 24 apresenta a tela de controle sendo acessada através de um *smartphone*.

Figura 24 – Sistema sendo acessado através de um smartphone



Fonte: Fotos da autora

O sistema inicialmente está funcionando com funcionalidades básicas, dispostas em quatro páginas, que são: inicial, visualização de imagens da câmera de monitoramento, controle de usuários e controle do sistema.

5.3.1 Página inicial

A página inicial do protótipo Domus, contém algumas informações a respeito do sistema e local para realizar o *login* com usuário e senha pré-cadastrado pelo administrador, conforme se pode visualizar na imagem a seguir.

Figura 25 – Página Principal Domus Automação Residencial



Fonte: Adaptado pela autora

A página principal é a que fornece acesso ao sistema. Quando o usuário acessá-la, deverá informar qual o seu *login* e senha, caso seja um administrador, assim que clicar no botão enviar, será exibida a página de administrador, se for um usuário comum do sistema, a página de administração do protótipo será exibida.

5.3.1.1 Segurança da página

Um dos fatores mais importantes quando se trata de desenvolvimento Web, está ligado à segurança.

Os avanços tecnológicos têm possibilitado cada vez mais o surgimento de novas técnicas de ataque aos sistemas, o que exige que haja um mínimo de segurança neles.

Principalmente em um sistema que fornece controle de uma residência, a necessidade de segurança torna-se ainda maior, portanto foi necessário adotar algumas medidas de proteção.

A primeira foi para evitar *SQL injection*, que ocorre quando um possível invasor insere instruções SQL num campo de formulário, *query string*, ou campo oculto, com a intenção de obter dados sigilosos.

Outra forma de evitar uma possível invasão, é utilizar o método POST para enviar informações, pois ele transmite as informações de forma mais protegida, dificultando o acesso aos usuários.

A criptografia da senha de usuários também é de grande importância, pois mesmo que haja uma invasão ao sistema de banco de dados, o invasor não consegue saber as senhas para acesso ao sistema.

Existem diversos algoritmos de criptografia, como md5, sha1, sha512, base64, Salsa20, *Whirlpool*, entre outros. O que foi utilizado no desenvolvimento desse sistema foi método de criptografia sha1.

O sha1 é um algoritmo de criptografia que gera uma sequência binária de 20 bytes ou 40 símbolos hexadecimais, portanto as senhas dos usuários ficarão salvas criptografadas no banco de dados.

5.3.2 Página de gerenciamento de usuários

A página de gerenciamento de usuários permite acesso somente dos administradores do sistema. Através dela, é possível realizar o cadastramento e exclusão de usuários do sistema.

Figura 26 – Página de gerenciamento de usuários



Formulário para cadastrar usuários

Nome: E-mail:

Login: Senha:

usuario

Usuários			
Nome	E-mail	Login	Excluir
arduino	arduino@gmail.com	arduino	sim
admin	admin	admin	sim

Fonte: Adaptado pela autora

Caso o administrador desejar cadastrar um novo usuário, basta inserir os dados como nome, e-mail, *login*, senha e selecionar o perfil, e clicar em salvar.

Caso desejar excluir um usuário cadastrado, basta clicar em sim, que a exclusão será realizada.

5.3.3 Página de controle do sistema

A página de controle do sistema é a principal do protótipo, pois é através dela que será realizado todo o gerenciamento dos dispositivos.

Assim que ela for acessada, abrirá uma comunicação com o Aduino através do método `socket_connect`, onde estão inseridas as configurações de IP e porta para acesso à plataforma de protipagem eletrônica, conforme se verifica no trecho de código a seguir.

Figura 27 – Função `socket_connect`

```

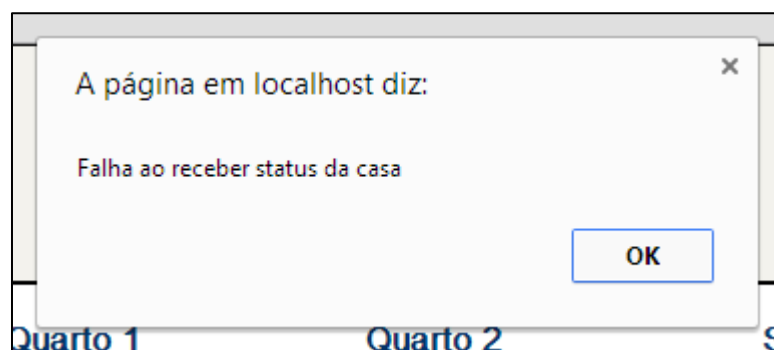
25 <?php
26
27 $sock = socket_create(AF_INET, SOCK_STREAM, SOL_TCP) or die("Unable to create socket\n");;
28 socket_connect($sock, "192.168.1.102", "8083");// Se conecta ao IP e Porta:
29

```

Fonte: Adaptado pela autora

Ela é composta por botões de controle, que serão carregados somente caso a comunicação com a central de automação (Arduino) estiver acessível, se houver algum erro de comunicação, seja por indisponibilidade da rede, ou qualquer outro fator, será exibido somente uma mensagem de erro, conforme se verifica a Figura 28.

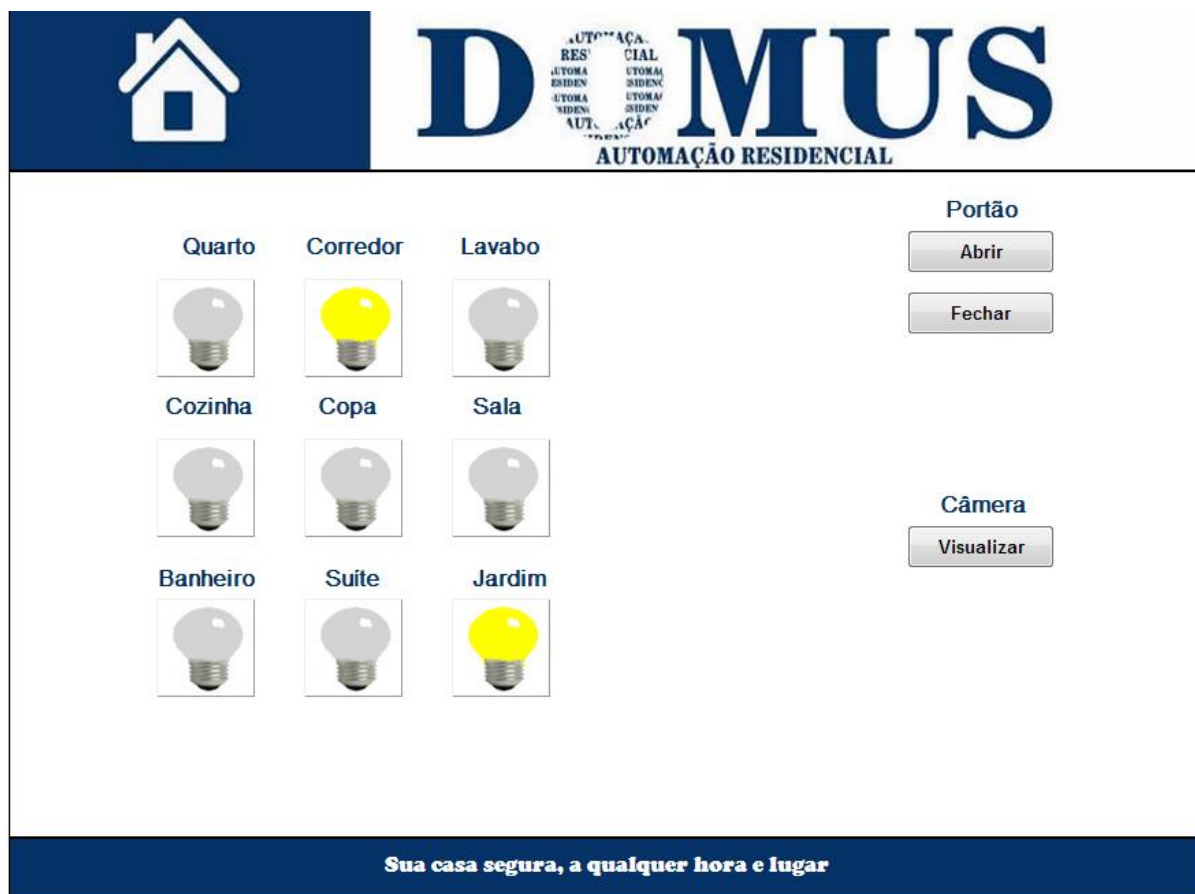
Figura 28 – Mensagem de erro ao receber status da casa



Fonte: Adaptado pela autora

Se não houver nenhum problema, a página irá carregar todos os seus componentes, conforme se visualiza a seguir.

Figura 29 – Tela de Controle do sistema



Fonte: Adaptado pela autora

Cada um dos botões exibidos na Figura 29 possui uma função, como acender ou apagar as lâmpadas, visualizar imagens da câmera de segurança e abrir ou fechar o portão. Nos tópicos a seguir, será explicado o que ocorre ao clicar em cada um destes.

5.3.3.1 Botão de controle das lâmpadas

Quando o usuário clicar em um dos botões de controle de lâmpadas, a página Web enviará uma mensagem contendo bit zero ou bit um, ao módulo *Ethernet*, sendo zero para desligar e um para acender a lâmpada.

No trecho de código a seguir, verifica-se como foi feito.

Figura 30 – Código mensagens enviadas pela página PHP

```

34 if(isset($_POST['bits'])) {
35     $msg = $_POST['bits'];
36     if(isset($_POST['Fora'])) { if($msg[0]=='1') { $msg[0]='0'; } else { $msg[0]='1'; }}
37     if(isset($_POST['Sala'])) { if($msg[1]=='1') { $msg[1]='0'; } else { $msg[1]='1'; }}
38     if(isset($_POST['Cozinha'])) { if($msg[2]=='1') { $msg[2]='0'; } else { $msg[2]='1'; }}
39     if(isset($_POST['Copa'])) { if($msg[3]=='1') { $msg[3]='0'; } else { $msg[3]='1'; }}
40     if(isset($_POST['Suite'])) { if($msg[4]=='1') { $msg[4]='0'; } else { $msg[4]='1'; }}
41     if(isset($_POST['Banheiro'])) { if($msg[5]=='1') { $msg[5]='0'; } else { $msg[5]='1'; }}
42     if(isset($_POST['Lavabo'])) { if($msg[6]=='1') { $msg[6]='0'; } else { $msg[6]='1'; }}
43     if(isset($_POST['Quarto'])) { if($msg[7]=='1') { $msg[7]='0'; } else { $msg[7]='1'; }}

```

Fonte: Adaptado pela autora

Os valores enviados através da variável \$msg serão interpretados pelo módulo *Ethernet*, e guardados no vetor msg[11], declarado no código fonte do Arduino, conforme verifica-se abaixo.

Figura 31 – Vetor declarado no código fonte do Arduino

```

// String onde é guardada as msgs recebidas
char msg[11] = "00000000L#";

```

Fonte: Adaptado pela autora

O Arduino está programado para ler bit a bit da *string*, por exemplo, se a *string* recebida da página Web for “11000000L#”, as duas primeiras lâmpadas serão acesas, se for “00000111L#” as três últimas serão acesas, e assim sucessivamente.

5.3.3.2 Botões de controle do portão

Para realizar o controle do portão, existem dois botões, um para abrir e outro para fechar. Quando o usuário clicar no botão abrir, a página Web enviará uma mensagem com o comando G#.

Figura 32 – Comando enviado pela página PHP para

```

45 if(isset($_POST['Abrir'])) { $msg = 'G#'; }

```

Fonte: Adaptado pela autora

Quando o Arduino receber a mensagem P#, ele enviará o pulso para a porta digital 13, o que irá fazer o servo motor se deslocar da posição 0°, até 180° que é a posição do portão aberto, de 1° em 1°, com um atraso de 25 milissegundos, que é o tempo necessário dele se deslocar até a posição programada.

Se o botão acionado for o de fechar, a página Web enviará a mensagem P# para o Arduino.

Figura 33 – Comando enviado pela página PHP para fechar portão

```
44 if(isset($_POST['Fechar'])) { $msg = 'P#'; }
```

Fonte: Adaptado pela autora

Quando a mensagem for recebida, o Arduino enviará o pulso para a porta onde está o servomotor, e o fará deslocar da posição 180° até sua posição inicial de 0°, com um atraso de 25 milissegundos.

No trecho de código a seguir desenvolvido para o Arduino, pode-se verificar melhor como foi realizada essa programação.

Figura 34 – Programação servo motor

```
case 'G':

    if (contagem < 50){ // Condição abrir, se a variavel contagem for menor que 20 faz o código que esta ente {}
    for(posicao = 90; posicao < 180; posicao +=1) // Servo gira da posição 0 para a posição 180 de 1 em 1 grau
    {
        servol.write(posicao);          // "Escreve" no pino 13 a posição para onde o servo se vai deslocar
        delay(25);                     // Espera de 25 milissegundos para que o servo possa chegar á sua posição
    }

    contagem ++;                       // Adiciona mais 1 á variavel contagem
    }

    case 'P':
        if (contagem < 50){
        for(posicao = 180; posicao >= 90; posicao -=1) // Servo gira da posição 180 para a posição 0 de 1 em 1 grau
        {
            servol.write(posicao);        // "Escreve" no pino 13 a posição para onde o servo se vai deslocar
            delay(25);                   // Espera de 25 milissegundos para que o servo possa chegar á sua posição
        }
        contagem ++;                     // Adiciona mais 1 á variavel contagem
        }

        break;
```

Fonte: Adaptado pela autora

5.3.3.3 Botões de visualização das imagens da câmera

Quando o usuário clicar no botão visualizar da página de controle do sistema, será exibida a página de visualização de imagens da câmera, descrita no tópico a seguir.

5.3.4 Página de visualização de imagens da câmera

A visualização de imagens será exibida quando o usuário do sistema clicar no botão visualizar da página de controle do sistema.

Através dela é possível controlar a direção da câmera, com movimentação de 270° horizontal e 90° vertical, resolução da imagem, frames por segundo, acompanhamento das imagens, capturar imagens, e configurar a detecção de movimento, sendo que, sempre que for verificada a presença de movimentos nas proximidades, e-mails são enviados com a imagem que a câmera capturou.

Figura 35 – Página de visualização de imagens



Fonte: Adaptado pela autora

Possui basicamente um *iframe*, que contém a página original de gerenciamento da câmera.

Um *iframe* torna possível a exibição de uma página dentro de outra, ou seja, neste caso, a página original de controle da câmera está sendo exibida dentro da página de acesso às imagens do protótipo.

6 CONCLUSÃO E TRABALHOS FUTUROS

Esse documento apresentou um pouco do universo da domótica, uma nova tecnologia que está em fase de grande crescimento no Brasil.

Verificaram-se os sistemas de domótica já existentes, quais suas características, benefícios proporcionados, mercado de consumidores, entre outros fatores. O que se notou mais relevante foi o alto custo dessa tecnologia, portanto o presente trabalho apresentou uma nova forma de automatizar residências, com uma tecnologia de código aberto, o que pode deixar o sistema com custo muito inferior aos existentes no mercado.

Portanto, realizou-se o desenvolvimento de um protótipo de um sistema de automação residencial, denominado Domus Automação Residencial, que integrou diversas tecnologias de grande reconhecimento no mercado, como a linguagem de programação PHP, servidor Web Apache, banco de dados MySQL e a plataforma Arduino, desenvolvida a poucos anos, mas não menos reconhecida.

Por se tratar de um tema novo, as principais dificuldades encontradas foram relacionadas aos referenciais teóricos, poucos autores escreveram sobre essa área, o que tornou um pouco difícil localizar especificações, necessidades, e algumas características dos sistemas de domótica.

Apesar das dificuldades citadas, os resultados do desenvolvimento do protótipo foram satisfatórios, principalmente porque tornou possível aprofundar conhecimento em diversas tecnologias, pois foi necessário aprender na prática como configurar um servidor Web, configurar roteadores, desenvolver sistemas e principalmente sobre programação e configurações do Arduino, que sempre despertou grande curiosidade.

Ao fim deste trabalho, verifica-se que é sim possível ter no mercado sistemas de automação, cujo desenvolvimento é simples, rápido e com tecnologias livres.

Através desse documento, pode-se desmistificar a ideia que desenvolver sistemas de controle, como o que foi apresentado é algo complicado e, possivelmente despertar o interesse acadêmico em uma área tão pouco difundida no país.

6.1 Trabalhos futuros

Durante o desenvolvimento do protótipo verificou-se que existem diversas melhorias que podem ser realizadas, tais como:

- Melhoria da interface do sistema Web, tornando-a mais amigável e intuitiva;
- Automação de outros dispositivos e componentes de uma residência, como sistema de climatização, cortinas e persianas, sons ambientes, temperatura do banho, irrigação de jardins, entre outros;
- Adição de uma tecnologia que seja capaz de prover controle através de comandos de voz;
- Adição de novas funcionalidades, como o envio de mensagens ao celular do proprietário caso ocorra alguma situação pré-configurada; e
- Tornar o protótipo um sistema comercializável.

REFERÊNCIAS

- ARDUINO. **Ethernet Library**. Disponível em: < <http://arduino.cc/en/reference/ethernet> > Acesso em: 8 set. 2013.
- ARDUINO. **SPI Library**. Disponível em: < <http://arduino.cc/en/Reference/SPI> >. Acesso em: 8 set. 2013.
- ARDUINO. **Servo Library**. Disponível em: < <http://arduino.cc/en/Reference/servo> >. Acesso em 8 set. 2013.
- ARDUINO. **Ethernet Shield**. Disponível em: < <http://www.arduino.cc/en/Main/ArduinoEthernetShield> >. Acesso em 8 set. 2013.
- ARDUINO. **Arduino Mega 2560N**. Disponível em: < <http://arduino.cc/en/Main/arduinoBoardMega2560> >. Acesso em 8 set. 2013.
- AURESIDE. **Desmistificando a domótica**. Disponível em: < <http://www.aureside.org.br/artigos/default.asp?file=01.asp&id=74> >. Acesso em 8 set. 2013.
- BOLZANI, C.A.M. **Análise de Desenvolvimento de uma Plataforma para Residências Inteligentes**. Tese (Doutorado em Engenharia Elétrica) – Escola Politécnica, Universidade de São Paulo, São Paulo, 2010.
- BOLZANI, C.A.M. **Residências Inteligentes**. São Paulo: 1. ed. Livraria da Física, 2004.
- BRUNO, Odemir M.; ESTROZI, Leandro F.; NETO, João E. S. Batista. **Programando Para a Internet com PHP**. Rio de Janeiro: Brasport, 2010.
- CARVALHO, Daniel Balparda de. **Segurança de Dados com Criptografia, Métodos e Algoritmos**. 2. ed. Rio de Janeiro: Book Express, 2001.
- CHAMUSCA, Alexandre. **Domótica & Segurança Electrónica**. Ingenium, 2006.
- CHOI, Wankyu; KENT, Allan; LEA Chris; PRASAD, Ganesh; ULLMAN, Chris. **Beginning PHP 4 Programando**. Tradução de Aldir José Coelho Correa da Silva e Flávia Cruz. São Paulo: Makron Books, 2001.
- COMUNICAÇÃO SPI com Arduino. 20 setembro 2012. Disponível em: < <http://labdegaragem.com/profiles/blogs/tutorial-comunica-o-spi-serial-peripheral-interface-com-arduino> >. Acesso em: 02 nov. 2013.
- DELISLE, Marc. **Mastering phpMyadmin 3.4 for Effective MySQL Management**. Disponível em: < <http://books.google.com.br/books?id=m1cUk-zcyOAC&printsec=frontcover&dq=phpmyadmin&hl=pt-BR&sa=X&ei=Mml->

UpHuEoLesATj-4KoCg&ved=0CEAQ6AEwAQ#v=onepage&q=phpmyadmin&f=false
> Acesso em: 5 nov.

EXAME. **Automação residencial deve movimentar US\$ 250 bi em sete anos.**

Revista Exame, 05 setembro 2012. Disponível em: <

<http://exame.abril.com.br/tecnologia/noticias/automacao-residencial-deve-movimentar-us-250-bi-em-sete-anos-m0079161> >. Acesso em: 26 out.

FERREIRA, João Alexandre Oliveira. **Interface Homem Máquina Para Domótica Baseado em Tecnologias WEB**. 2008. Dissertação (Mestrado Integrado em Engenharia Electrotécnica e de Computadores) - Faculdade de Engenharia da Universidade do Porto, Porto, Portugal.

GDS Automação. **GDS Automação Residencial**. Disponível em: <

<http://www.gdsautomacao.com.br/site> >. Acesso em 26 out.

GRANDO, Nei. **O ciclo de adoção de Novos Produtos pelos Consumidores.**

Neigrando. 26 janeiro 2013. Disponível em: <

<http://neigrando.wordpress.com/2013/01/26/o-ciclo-de-adocao-de-novos-produtos-pelos-consumidores> >. Acesso em: 26 out.

IHOUSE. **Home controler**. Disponível em: < <http://ihouse.com.br> >. Acesso em: 13 out.

ILUFLEX. **Automação sem fio**. Disponível em: < <http://www.iluflex.com.br> >. Acesso em 13 out.

JÚNIOR, Adalberto Veloso; SIQUEIRA, Rodolfo Barbosa de. **Posicionamento de Câmera por Acelerômetro**. Disponível em: <

<http://www.ppgia.pucpr.br/~santin/ee/2008/3> >. Acesso em 21 set. 2013.

LAGUÁRDIA, Susy. **Automação residencial cresce no Brasil**. Disponível em: <

<http://www.areside.org.br/imprensa/default.asp?file=62.asp> >. Acesso em 22 set. 2013.

LANG, Mariana. **Sistemas de automação residencial transferem para tablet e celular o controle da casa**. Folha de São Paulo, 13 dezembro 2012. Disponível em:

<<http://www1.folha.uol.com.br/tec/1194228-sistemas-de-automacao-residencial-transferem-para-tablet-e-celular-o-controle-da-casa.shtml>>. Acesso em: 22 ago. 2013.

MCROBERTS, Michael. **Arduino Básico**. São Paulo: Novatec, 2011.

MCROBERTS, Michael. **Beginning Arduino**. United States of America: TIA, 2010.

MASLAKOWSKI, Mark. **Aprenda em 21 dias MySQL**. Tradução de Edson Furmankiewicz e Joana Figueiredo. Rio de Janeiro: Campus, 2000.

MOYA, J. M. Huidobro; TEJEDOR, R. J. Milán. **Manual de Domótica**. 2010. Disponível em: <
http://books.google.com.br/books?id=V6lzqgDcfF8C&printsec=frontcover&hl=pt-BR&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false>. Acesso em: 23 ago. 2013.

NETCRAFT. **November 2013 Web Server Survey**. Disponível em: <
<http://news.netcraft.com/archives/2013/11/01/november-2013-web-server-survey.html>>. Acesso em 29 set. 2013.

OLIVEIRA, J.P. **Domótica: Perspectiva da Plataforma Arduino**. 2012. Monografia (Conclusão de Curso) – Universidade Estadual de Goiás, Goianésia.

PIGHIXX. **Atmega 2560**. Disponível em: <
<http://www.pighixx.com/downloads/atmega2560>>. Acesso em: 02 nov. 2013.

SENA, D.C.S. **Automação Residencial**. 2005. Monografia (Conclusão de Curso) – Universidade Federal do Espírito Santo, Vitória.

SILVEIRA, Paulo Rogério da. **Automação e Controle Discreto**. 9. Ed. São Paulo: Érica, 1998.

SIMPLIFIES. **SimpleHome**. Disponível em: <
<http://www.simplifies.com.br/home/index.php>>. Acesso em: 26 out.

TENVIS. **Câmera JPT3815W 2013**. Disponível em: <
<http://www.tenvis.com/jpt3815w-2013-new-pantilt-wireless-ip-camera-night-vision-p-218.html>>. Acesso em: 23 ago. 2013.

TURUEL, Evandro Carlos. **Uma Proposta de Framework para Sistemas de Automação Residencial com Interface para Web**. 2008. Dissertação (Mestrado em Tecnologia: Gestão Desenvolvimento e Formação) - Centro Estadual de Educação Tecnológica Paula Souza, São Paulo, Brasil.